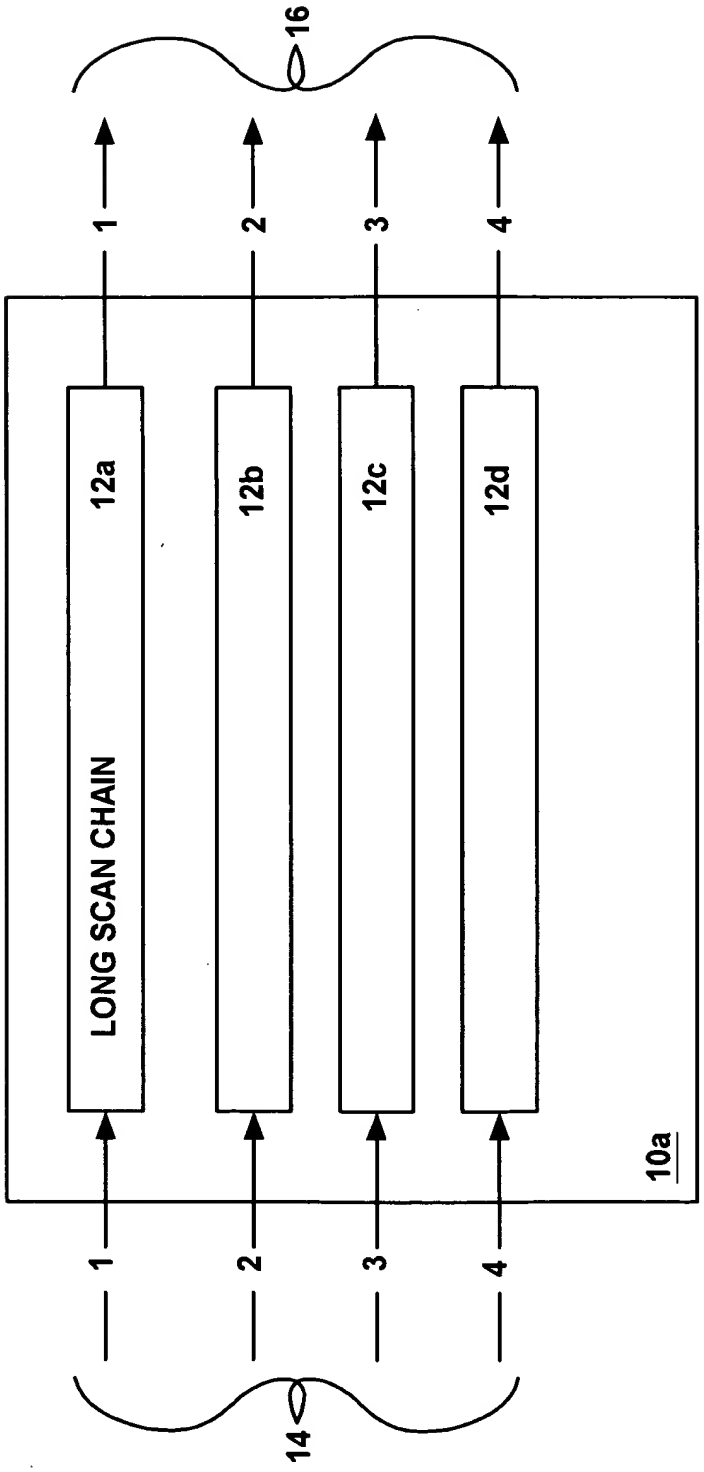

FIGURE 1A

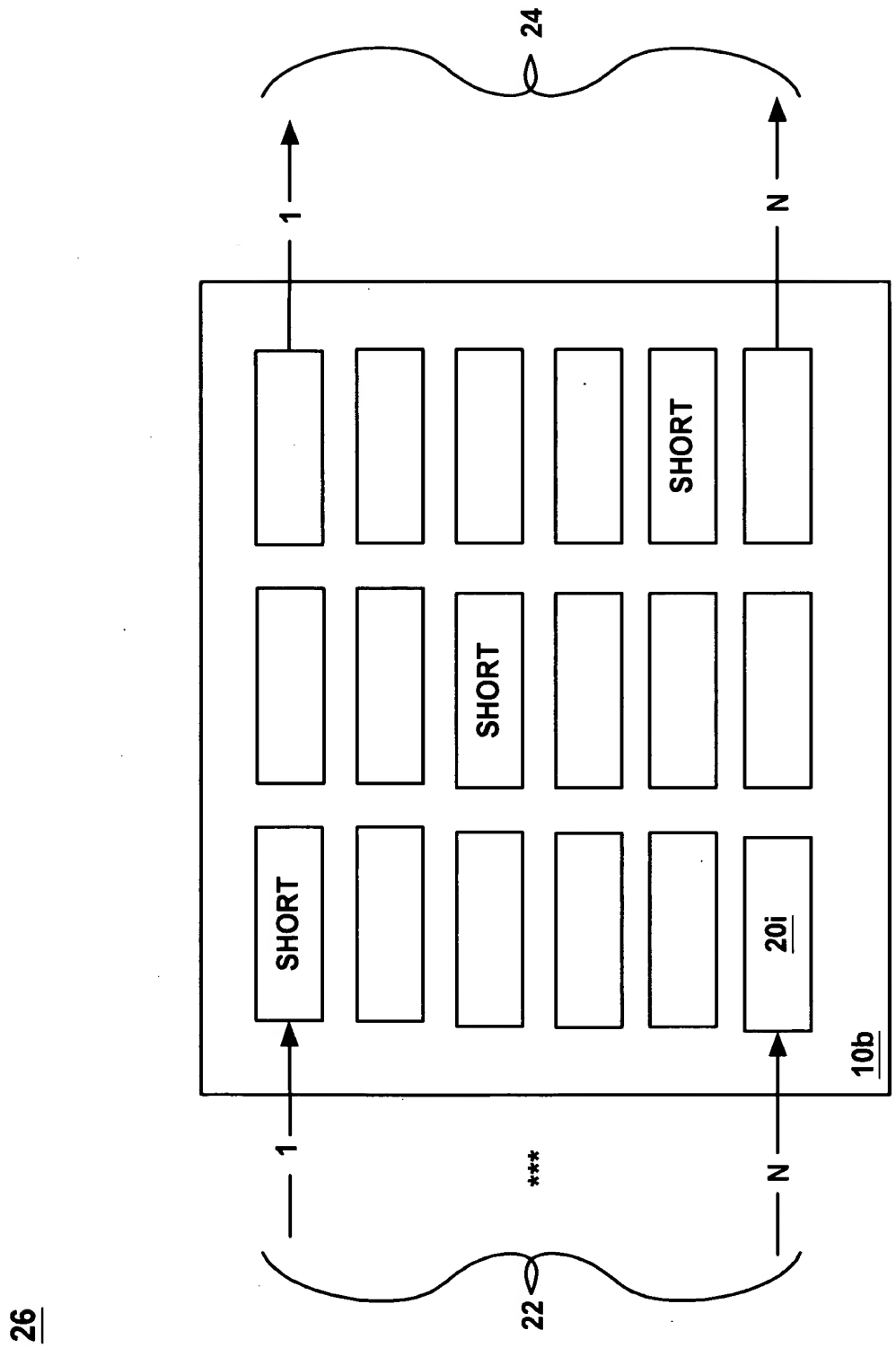

FIGURE 1B

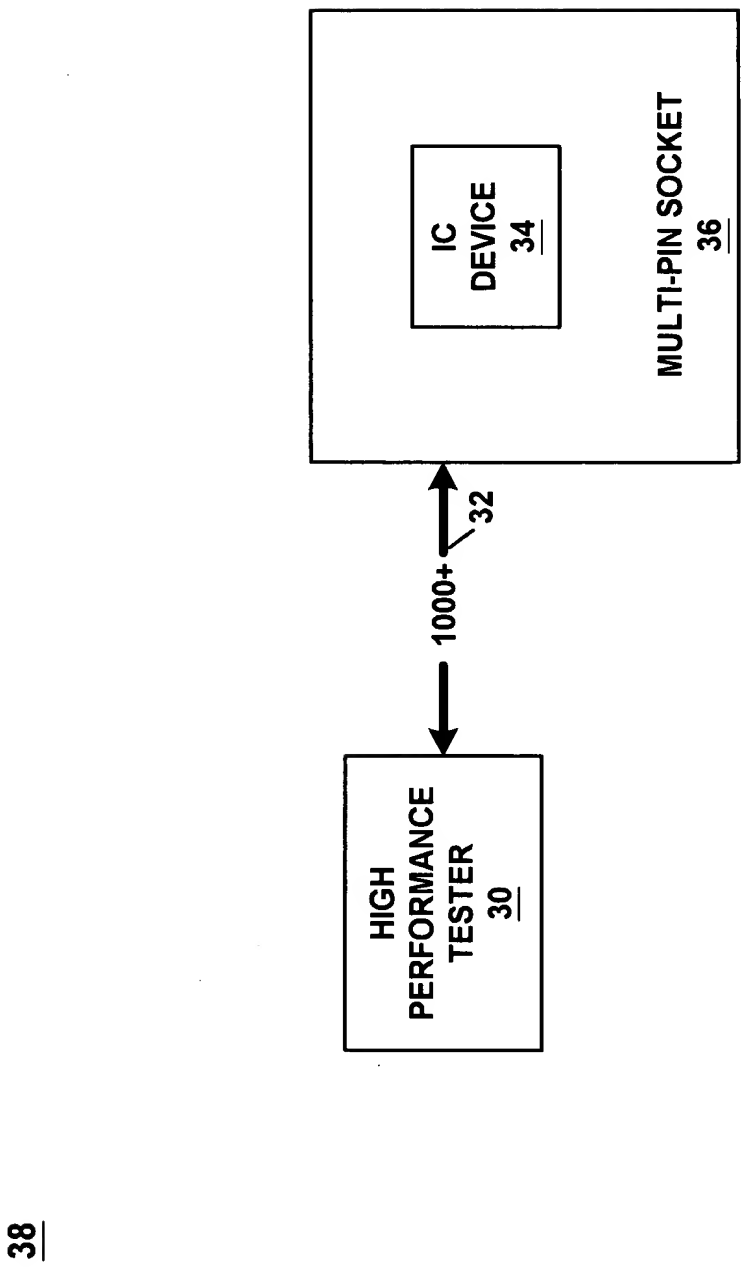
38
HIGH PERFORMANCE TESTER 30
1000+
32
IC DEVICE 34
MULTI-PIN SOCKET 36

FIGURE 1C

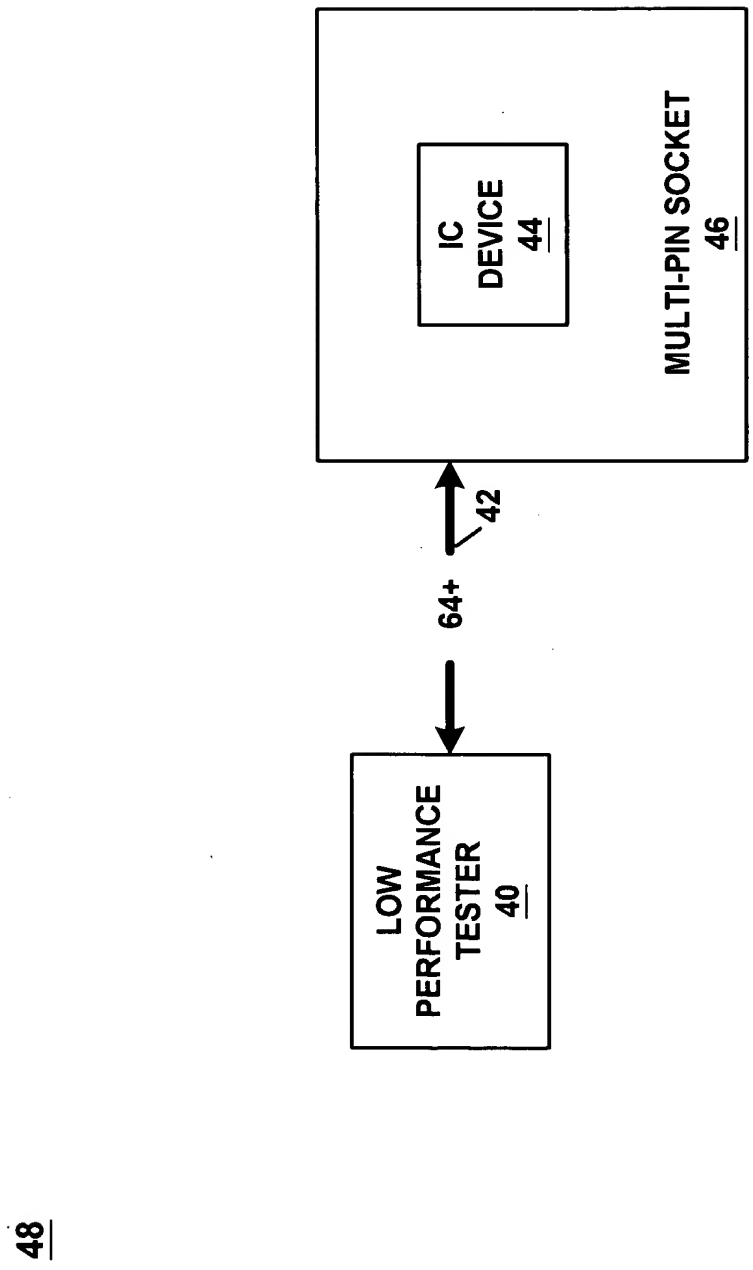
48
LOW PERFORMANCE TESTER 40
64+
42
IC DEVICE 44
MULTI-PIN SOCKET 46

FIGURE 1D

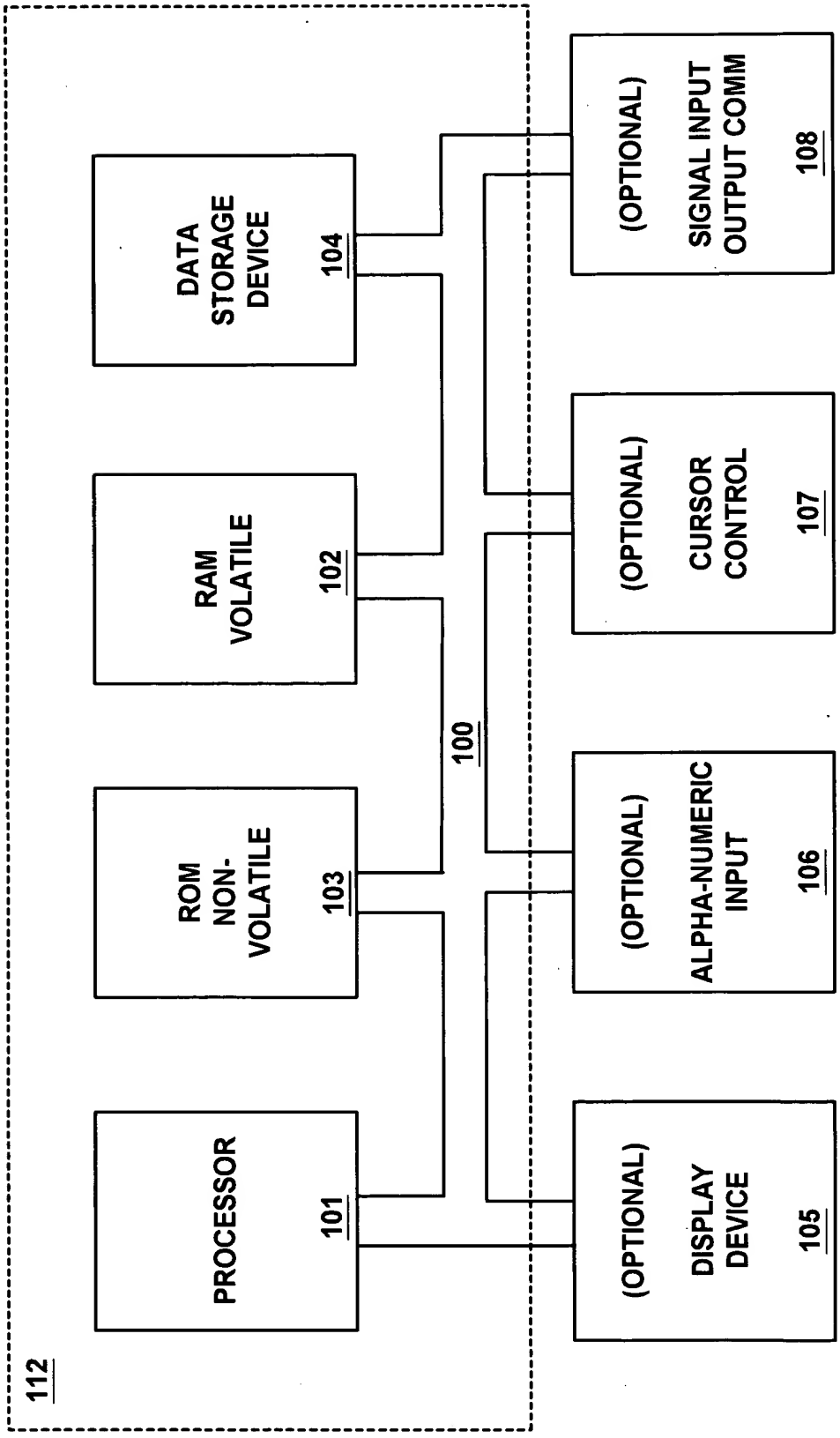
112
PROCESSOR
101
ROM
NON-
VOLATILE
103
RAM
VOLATILE
102
DATA
STORAGE
DEVICE
104
100
(OPTIONAL)
DISPLAY
DEVICE
105
(OPTIONAL)
ALPHA-NUMERIC
INPUT
106
(OPTIONAL)
CURSOR
CONTROL
107
(OPTIONAL)
SIGNAL INPUT
OUTPUT COMM
108

FIGURE 2

200

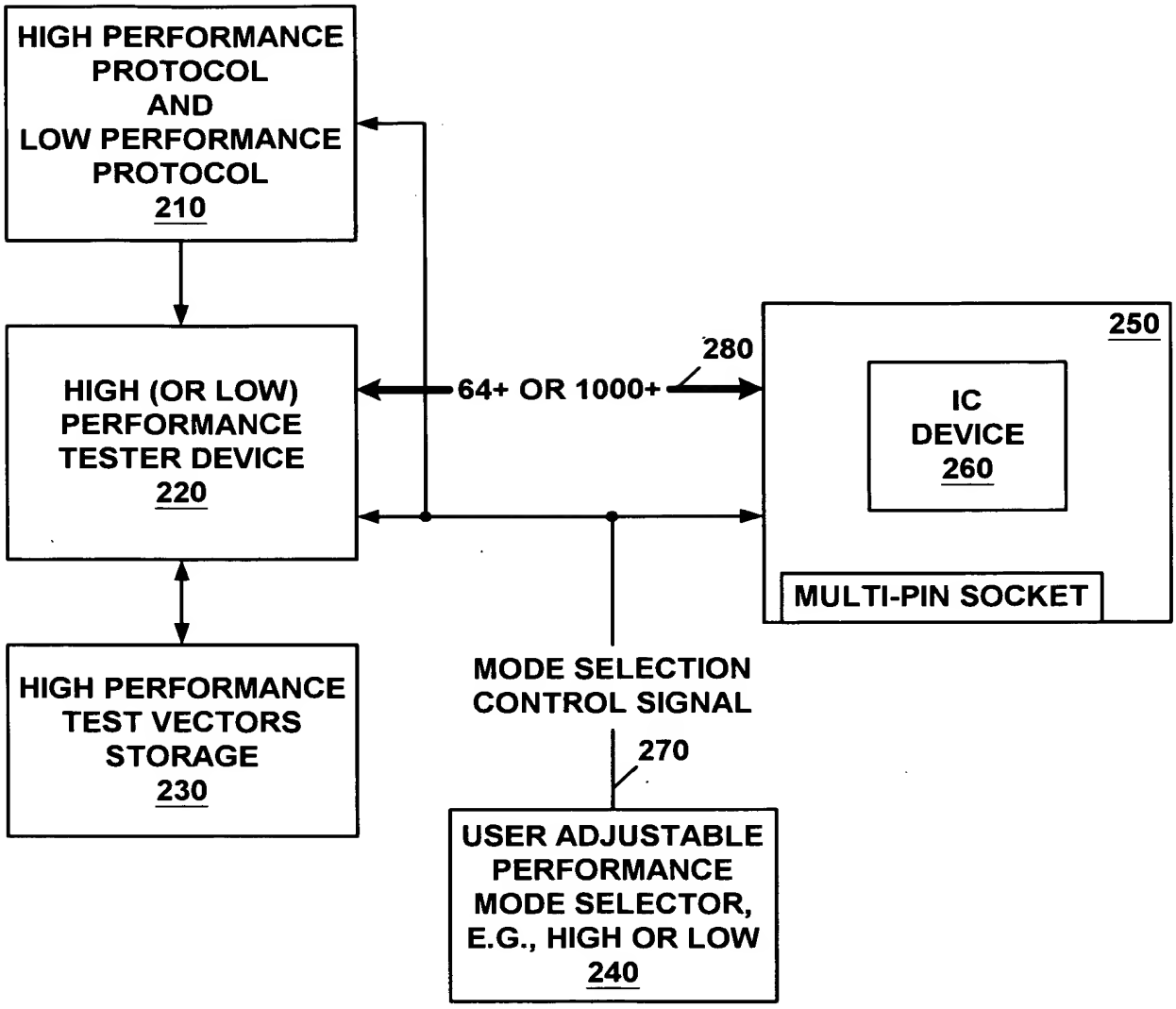
HIGH PERFORMANCE PROTOCOL AND LOW PERFORMANCE PROTOCOL 210
HIGH (OR LOW) PERFORMANCE TESTER DEVICE 220
HIGH PERFORMANCE TEST VECTORS STORAGE 230
64+ OR 1000+
280
250
IC DEVICE 260
MULTI-PIN SOCKET
MODE SELECTION CONTROL SIGNAL
270
USER ADJUSTABLE PERFORMANCE MODE SELECTOR, E.G., HIGH OR LOW 240

FIGURE 3

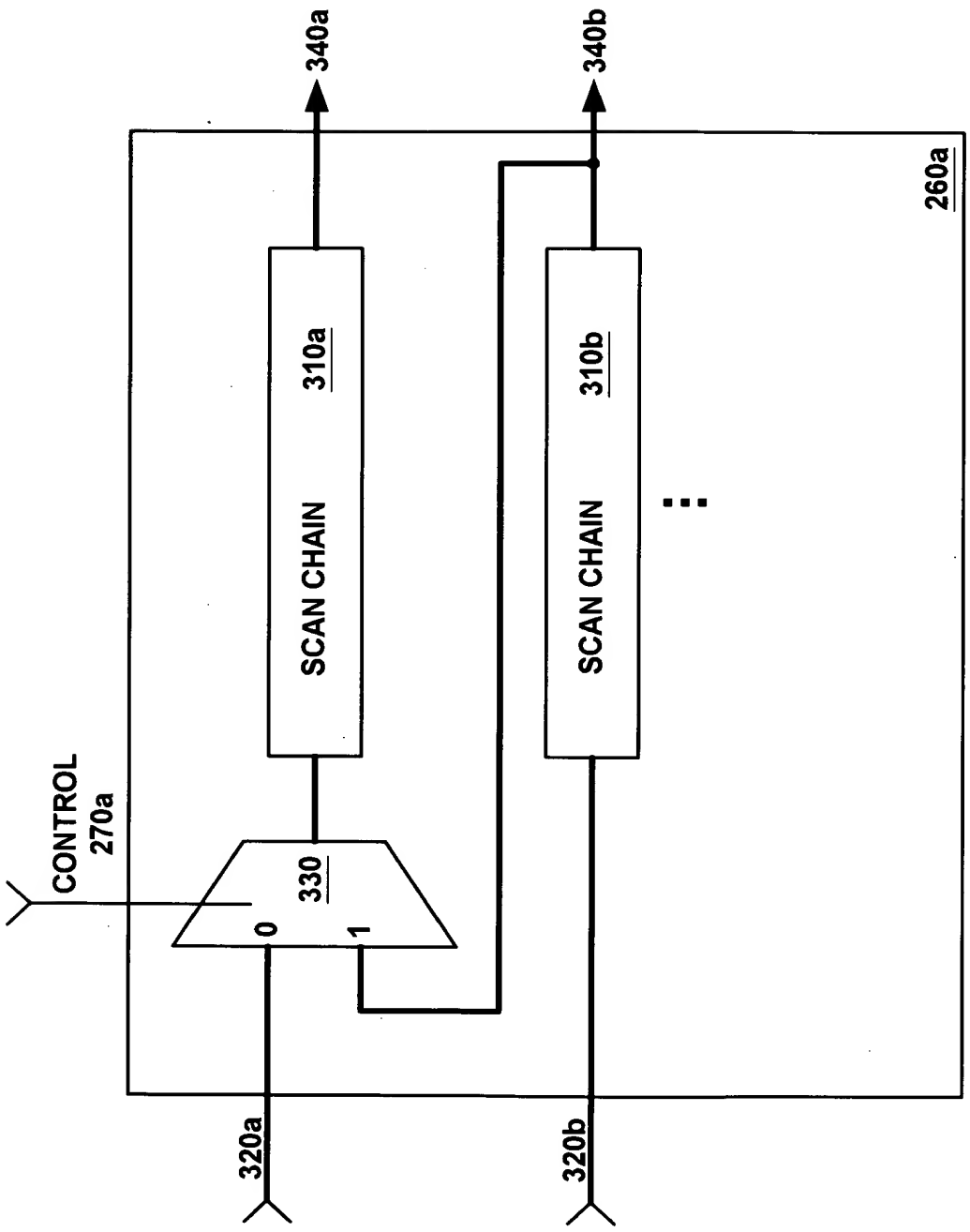
340a
340b
260a
SCAN CHAIN
310a
SCAN CHAIN
310b
CONTROL
270a
330
0
1
320a
320b

FIGURE 4A

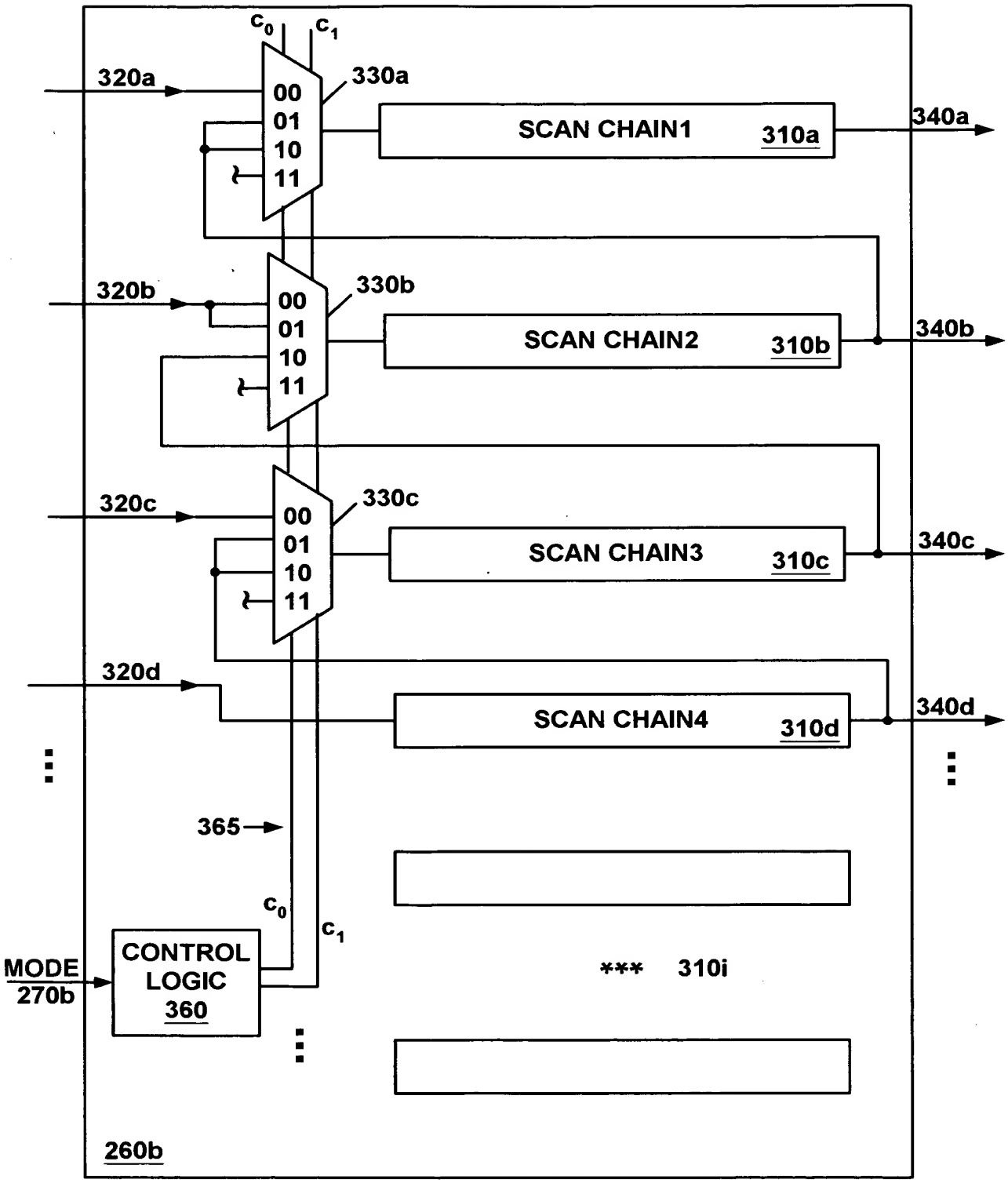
c_0
c_1
320a
330a
00
01
10
11
SCAN CHAIN1
310a
340a
320b
330b
SCAN CHAIN2
310b
340b
320c
330c
SCAN CHAIN3
310c
340c
320d
SCAN CHAIN4
310d
340d
365
c_0
c_1
MODE
270b
CONTROL
LOGIC
360
*** 310i
260b

FIGURE 4B

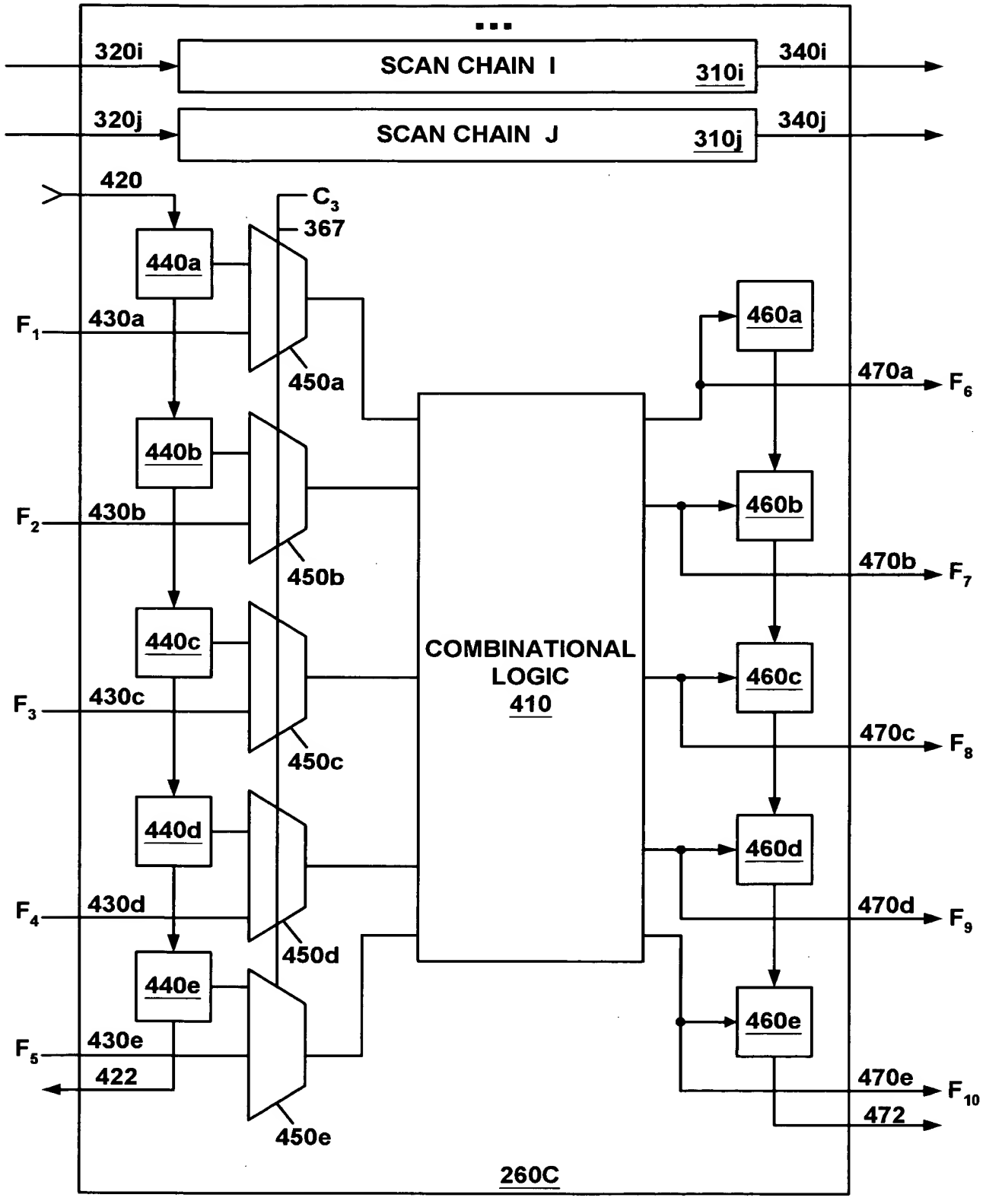
320i
SCAN CHAIN I
310i
340i
320j
SCAN CHAIN J
310j
340j
420
C_3
367
440a
F_1
430a
450a
460a
470a
F_6
440b
F_2
430b
450b
460b
470b
F_7
440c
F_3
430c
450c
COMBINATIONAL LOGIC
410
460c
470c
F_8
440d
F_4
430d
450d
460d
470d
F_9
440e
F_5
430e
422
450e
460e
470e
F_{10}
472
260C

FIGURE 5

500

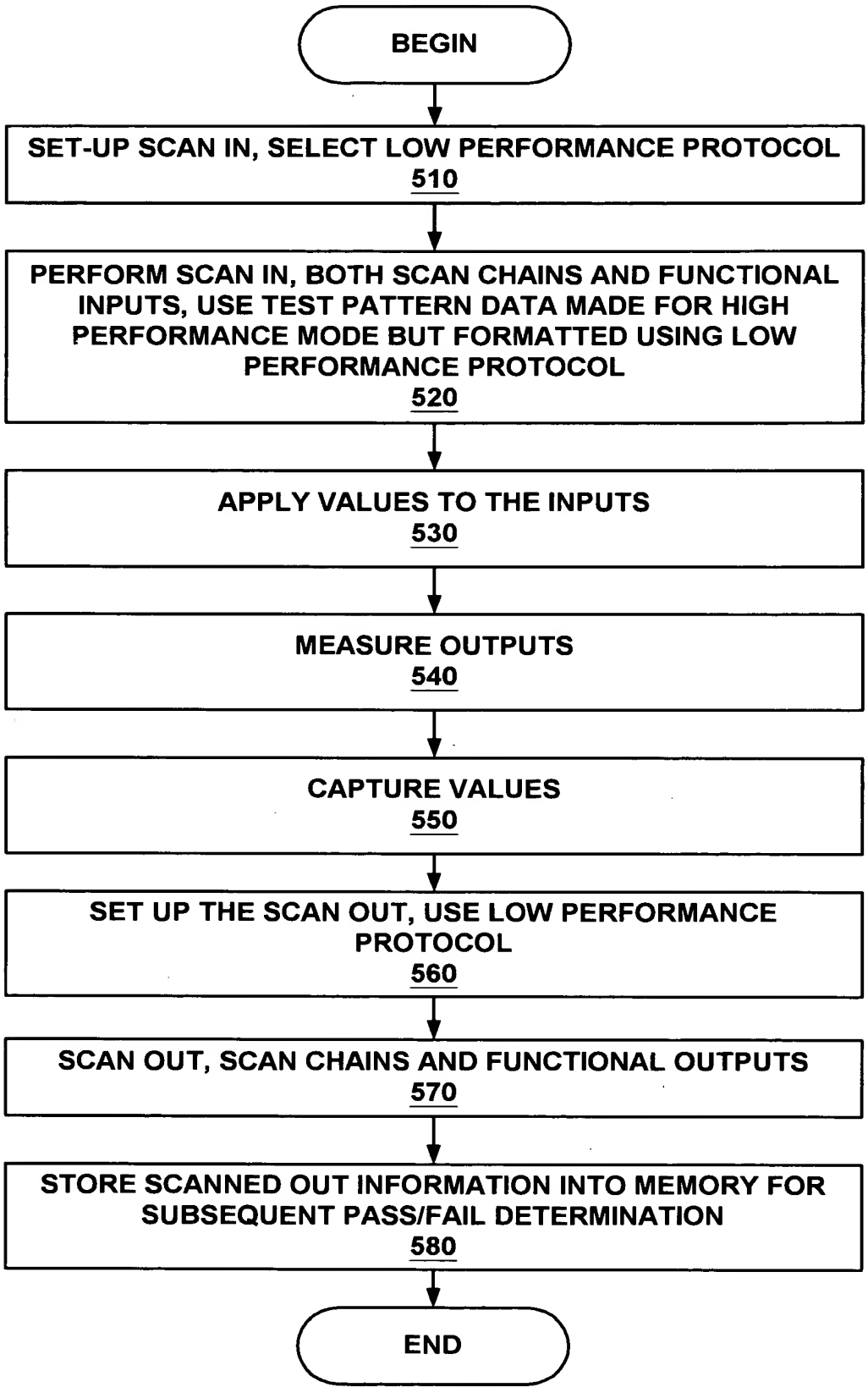
BEGIN
SET-UP SCAN IN, SELECT LOW PERFORMANCE PROTOCOL
510
PERFORM SCAN IN, BOTH SCAN CHAINS AND FUNCTIONAL INPUTS, USE TEST PATTERN DATA MADE FOR HIGH PERFORMANCE MODE BUT FORMATTED USING LOW PERFORMANCE PROTOCOL
520
APPLY VALUES TO THE INPUTS
530
MEASURE OUTPUTS
540
CAPTURE VALUES
550
SET UP THE SCAN OUT, USE LOW PERFORMANCE PROTOCOL
560
SCAN OUT, SCAN CHAINS AND FUNCTIONAL OUTPUTS
570
STORE SCANNED OUT INFORMATION INTO MEMORY FOR SUBSEQUENT PASS/FAIL DETERMINATION
580
END

FIGURE 6

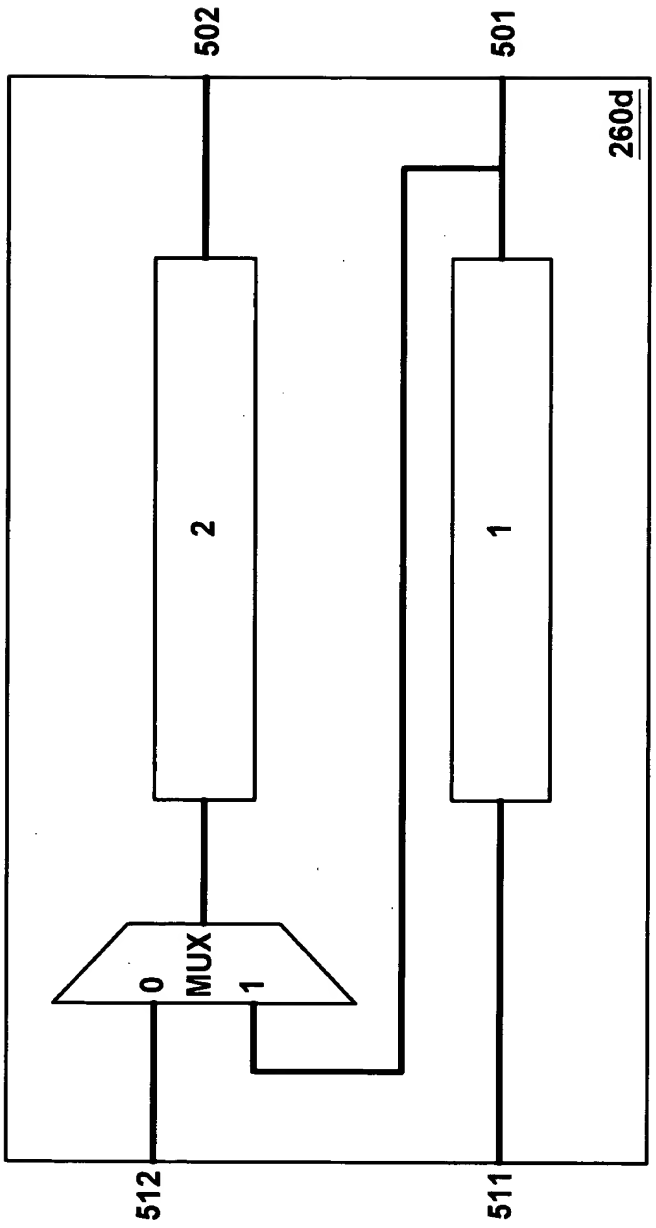

FIGURE 7A

600

```
MacroDefs {
        Do_one_test {
            W normal_timing;
            C { scan_enable = 1; control_inputs = 'control_inputs'; }
            If ( reconfig_signal == many_inputs )
           {
            640 → Shift { V { si1 = 'si1'; si2 = 'si2'; so1 = 'S01_p'; so2 = 'S02_p'; clk'=P;}}
610 ←              C { scan_enable = 0;}
                   V { func_inputs = 'func_inputs'; func_outputs = 'func_outputs'; } ← 650
            {
            If ( reconfig_signal == few _inputs )
            {
                   Shift { V { si1 = 'si1 si2'; so2 = 'so2 so1';  ← 660
620 ←                                  funcIn = 'func_inputs'; funcOut = 'func_outputs_p'clk = P;} ← 670
                    C { scan_enable =0; }
            {
            V { clk = P;} —680
          {
{
```

12/12

FIGURE 7B